1. CPP Program to add two numbers through Class and Object

```
#include<iostream.h>
#include<conio.h>

class sample
{
 int a;
 public:
 void getdata()
 {
  cin>>a;
 }

 void putdata()
 {
  cout<<a;
 }

 void adddata(sample t1,sample t2)
 {
  a=t1.a+t2.a;
 }
};
void main()
{
```

```
clrscr();
sample a1,a2,a3;
a1.getdata();
a2.getdata();
a3.adddata(a1,a2);
a3.putdata();
getch();
}
```

OUTPUT

```
5
7
12
```

2. CPP Program to find quotient and remainder

```
#include<iostream.h>
#include<conio.h>

int main()
{
 clrscr();
 int a,b;
 cout<<"Enter any two numbers for division"<<endl;
 cout<<"First enter Divident, then Diviser"<<endl;
 cin>>a>>b;
 cout<<"The Quotient is: "<<a/b<<endl;
 cout<<"and the Remainder is: "<<a%b;
 getch();
}
```

OUTPUT

```
Enter any two numbers for division
First enter Divident, then Diviser
5
2
The Quotient is: 2
and the Remainder is: 1
```

3. CPP Program for find size of different datatypes

```
#include<iostream.h>

#include<conio.h>


void main()

{

 clrscr();

 cout<<"The size of char(Character) Datatype is: "<<sizeof(char)<<endl;

 cout<<"The size of int(Integer) Datatype is: "<<sizeof(int)<<endl;

 cout<<"The size of float Datatype is: "<<sizeof(float)<<endl;

 getch();

}
```

OUTPUT

```
The size of char(Character) Datatype is: 1

The size of int(Integer) Datatype is: 2

The size of float Datatype is: 4
```

4. CPP Program to check whether a char is Vowel or Consonant

```
#include<iostream.h>

#include<conio.h>

void main()

{

 clrscr();

 char ch;

 cout<<"Enter any Character"<<endl;

 cin>>ch;

 if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' || ch=='A' || ch=='E' || ch=='I' ||
ch=='O' || ch=='U')

  cout<<"It is a Vowel"<<endl;

 else

  cout<<"It is a Consonant";

  getch();

}
```

OUTPUT

```
Enter any Character

W

It is a Consonant
```

5. CPP Program to find largest number among three numbers using turnery operator

```
#include<iostream.h>
#include<conio.h>
void main()
{
 clrscr();
 int a,b,c,L;
 cout<<"Enter any three numbers"<<endl;
 cin>>a>>b>>c;
 L=a>b?(a>c?a:c):(b>c?b:c);
 cout<<endl<<"The Largest number is: "<<L;
 getch();
}
```

OUTPUT

```
Enter any three numbers
100
10
500

The Largest number is: 500
```

6. CPP Program to find large number among three numbers using conditional operator

```
#include<iostream.h>
#include<conio.h>
void main()
{
 clrscr();
 int a,b,c,L;
 cout<<"Enter any three numbers"<<endl;
 cin>>a>>b>>c;
 if(a>b)
  if(a>c)
   L=a;
  else
   L=c;
 else
  if(b>c)
   L=b;
  else
   L=c;
 cout<<endl<<"The Largest number is: "<<L;
 getch();
}
```

OUTPUT

Enter any three numbers

1112

1110

9999

The Largest number is: 9999

7. CPP Program for quadratic equation

```
#include<conio.h>
#include<iostream.h>
#include<conio.h>
#include<math.h>
void roots();
int a,b,c,D;
void main()
{
 clrscr();
 cout<<"Enter a quadratic equation(ax^2 + bx + c=0) to find its roots"<<endl;
 cout<<"Enter the value of a,b and c respectively"<<endl;
 cin>>a>>b>>c;
 D=b*b-4*a*c;
 if(D==0) {
  cout<<endl<<"The equation has only one root";
  cout<<endl<<"Root is: ";
  roots();
 }
 else if(D>0)
 {
  cout<<endl<<"The equation has two real roots";
  cout<<endl<<"Roots are: ";
  roots();
 }
 else
```

```
  cout<<"The equation has imaginary roots";
getch();
}
void roots()
{
 int x1,x2;
 x1=(-b+sqrt(D))/(2*a);
 x2=(-b-sqrt(D))/(2*a);
 if(x1==x2)
  cout<<x1;
 else
  cout<<"x= "<<x1<<","<<x2;
 getch();
}
```

OUTPUT

Enter a quadratic equation(ax^2+ bx + c=0) to find its roots

Enter the value of a,b and c respectively

2

-12

17

The equation has two real roots

Roots are: x= 3,2

8. CPP Program for find the sum of first n natural numbers

```
#include<conio.h>
#include<iostream.h>

void main()
{
 clrscr();
 int i,n,sum=0;
 cout<<"Enter the value of n for find the sum of first n natural numbers"<<endl;
 cin>>n;
 for(i=1;i<=n;i++)
  sum=sum+i;
 cout<<endl<<"The sum of first "<<n<<" natural numbers is: "<<sum;
 getch();
}
```

OUTPUT

```
Enter the value of n for find the sum of first n natural numbers
7

The sum of first 7 natural numbers is: 28
```

9. CPP Program for print Fibonacci Series using Recursion

```
#include<iostream.h>
#include<conio.h>
long int fibo(int n);
void main()
{
 int num,i;
 clrscr();
 cout<<"Enter nth term for finding FIBONACCI Series till nth term\n";
 cin>>num;
 for(i=1;i<=num;i++)
  cout<<fibo(i)<<"\t";
getche();
}

long int fibo(int n)
{
 if(n==1)
  return 0;
 if(n==2)
  return 1;
 else
  return(fibo(n-1)+fibo(n-2));
}
```

OUTPUT

Enter nth term for finding FIBONACCI Series till nth term

30

0 1 1 2 3 5 8 13 21 34

55 89 144 233 377 610 987 1597 2584 4181

6765 10946 17711 28657 46368 75025 121393 196418 317811 514229

10. CPP Program to display prime numbers between two intervals

```
#include<iostream.h>

#include<conio.h>

#include<math.h>


void main()

{

 clrscr();

 int n1,n2,i,j;

 cout<<"Enter the lower and minimum value for print prime numbers between
them"<<endl;

 cin>>n1>>n2;

 cout<<endl;

 for(i=n1;i<=n2;i++)

 {

  for(j=2;j<i;j++)

   if(i%j==0)

    break;

  if(j==i)

   cout<<j<<" ";

 }

 getch();

}
```

OUTPUT

Enter the lower and minimum value for print prime numbers between them

2

66

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61

11. CPP Program to display Armstrong number between two intervals

```
#include<iostream.h>
#include<conio.h>
#include<math.h>
int size(int);
void main()
{
 clrscr();
 cout<<"Enter the lowest and upper value to display Armstrong number between them"<<endl;
 int lower,upper,i,count,temp,remainder,sum;
 cin>>lower>>upper;

 for(i=lower;i<=upper;i++)
 {
  temp=i,sum=0;
  count=size(temp);
  while(temp!=0)
  {
   remainder=temp%10;
   sum=sum+pow(remainder,count);
   temp/=10;
  }
  if(sum==i)
   cout<<i<<" ";
 }
```

```
getch();
}

int size(int temp)
{
 int count=0;
 while(temp!=0)
 {
  temp/=10;
  count++;
 }
 return count;
}
```

OUTPUT

```
Enter the lowest and upper value to display Armstrong number between them
1
10000
1 2 3 4 5 6 7 8 9 153 370 371 407 1634 8208 9474
```

12. CPP Program for simple Calculator

```
#include<iostream.h>
#include<conio.h>

void main()
{
 clrscr();
 int a,b;
 char ch;
 cout<<"Enter any two numbers for calculations"<<endl;
 cout<<"Enter first operand, operator and second operand respectively"<<endl;
 cin>>a>>ch>>b;
 switch(ch)
 {
  case '+':
  cout<<a+b;
  break;

  case '-':
  cout<<a-b;
  break;

  case '*':
  cout<<a*b;
  break;
```

```
 case '/':
 cout<<(float)a/b;
 break;

 default:
 cout<<endl<<"Please Enter valid Operator";
 }
getch();
}
```

OUTPUT

```
Enter any two numbers for calculations
Enter first operand, operator and second operand respectively
5/2
2.5
```

13. CPP Program for reverse a sentence

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
#include<string.h>
void display(int,int,char[]);

void main()
{
 clrscr();
 char a[40];
 cout<<"Enter any Setence\n";
 gets(a);
 int length=strlen(a);

 int sti,endi;
 sti=length-1;
 endi=length-1;
 for(int i=length-1;i>=0;i--)
  if(a[i]==32 || i==0)
   {
    if(i==0)
     sti=i;
    else
     sti=i+1;
    display(sti,endi,a);
```

```
   endi=i-1;
  }
 getch();
}

void display(int sti,int endi,char str[])
{
 for(int i=sti;i<=endi;i++)
  cout<<str[i];
  cout<<" ";
}
```

OUTPUT

```
Enter any Sentence
India is great
great is India
```

14. CPP Program for find largest element in an array

```
#include<iostream.h>
#include<conio.h>

void main()
{
 clrscr();
 int a[50],i,size,largest;
 cout<<"Enter the number of elements"<<endl;
 cin>>size;
 cout<<endl<<"Now Enter "<<size<<" "<<"elements"<<endl;
 for(i=0;i<size;i++)
  cin>>a[i];

 largest=a[0];
 for(i=0;i<size;i++)
  if(a[i]>largest)
   largest=a[i];
 cout<<"The largest number is: "<<largest;
 getch();
}
```

OUTPUT

Enter the number of elements

7

Now Enter 7 elements

11

21

1

43

666

789

543

The largest number is: 789

15. CPP Program for call by value

```
#include<iostream.h>
#include<conio.h>
void swap(int,int);
void main()
{
 clrscr();
 int a,b;
 cout<<"Enter any two numbers for swapping"<<endl;
 cin>>a>>b;
 cout<<"You have entered\na= "<<a<<endl<<"b= "<<b;
 swap(a,b);
 getch();
}

void swap(int x,int y)
{
 x=x+y;
 y=x-y;
 x=x-y;

 cout<<endl<<"After swapping the numbers are"<<endl;
 cout<<"a= "<<x<<endl<<"b= "<<y;
}
```

OUTPUT

Enter any two numbers for swapping

12

78

You have entered

a= 12

b= 78

After swapping the numbers are

a= 78

b= 12

16. CPP Program for call by reference

```
#include<iostream.h>
#include<conio.h>
void swap(int &,int &);
void main()
{
 clrscr();
 int a,b;
 cout<<"Enter any two numbers for swapping"<<endl;
 cin>>a>>b;
 cout<<"You have entered\na= "<<a<<endl<<"b= "<<b;
 swap(a,b);
 cout<<endl<<"After swapping the numbers are"<<endl;
 cout<<"a= "<<a<<endl<<"b= "<<b;
 getch();
}

void swap(int &x,int &y)
{
 x=x+y;
 y=x-y;
 x=x-y;
}
```

OUTPUT

Enter any two numbers for swapping

111

222

You have entered

a= 111

b= 222

After swapping the numbers are

a= 222

b= 111

17. CPP Program for inline function

```
#include<iostream.h>
#include<conio.h>

class sample
{
 int a;

 public:
 void setdata(int x)
 {
  a=x;
 }
 inline void showdata();
};

void sample :: showdata()
{
  cout<<endl<<"a= "<<a;
}

void main()
{
 clrscr();
 sample s;
 int t;
```

```
 cout<<"Enter any data"<<endl;
 cin>>t;
 s.setdata(t);
 cout<<endl<<"You have entered";
 s.showdata();
 getch();
}
```

OUTPUT

```
Enter any data
123

You have entered
a= 123
```

18. CPP Program for function overloading

```
#include<iostream.h>
#include<conio.h>
int volume(int);   //Cube
double volume(double,int);     //  Cylinder
int volume(int,int,int);      //   Cuboid

void main()
{
 clrscr();
 cout<<"The Volume of Cube is: "<<volume(5)<<endl;
 cout<<"The Volume of Cylinder is: "<<volume(2.4,5)<<endl;
 cout<<"The Volume of Cuboid: "<<volume(2,4,5)<<endl;
 getch();
}
int volume(int a)
{
 return (a*a*a);
}

double volume(double a,int b)
{
 return(3.14*a*a*b);
}

int volume(int a,int b,int c)
```

```
{
 return(a*b*c);
}
```

OUTPUT

```
The Volume of Cube is: 125
The Volume of Cylinder is: 90.432
The Volume of Cuboid: 40
```

19. CPP Program to illustrate static data member

```
#include<iostream.h>
#include<conio.h>

class sample
{
 int a;
 static int count;

 public:
 void setdata()
 {
  a=++count;
 }

 void showdata()
 {
  cout<<"a= "<<a<<endl;
 }

 void showcount()
 {
  cout<<"count= "<<count<<endl;
 }
};
```

```
int sample :: count=90;


void main()
{
 clrscr();
 sample s1,s2,s3;
 s1.setdata();
 s2.setdata();
 s3.setdata();
 s1.showdata();
 s2.showdata();
 s3.showdata();
 s1.showcount();
 s2.showcount();
 s3.showcount();
 getch();
}
```

OUTPUT

```
a= 91
a= 92
a= 93
count= 93
count= 93
count= 93
```

20. CPP Program to illustrate static member function

```
#include<iostream.h>
#include<conio.h>
class sample
{
 int a;
 static int count;

 public:
 void setdata()
 {
  a=++count;
 }

 void showdata()
 {
  cout<<"a= "<<a<<endl;
 }

 static void showcount()
 {
  cout<<"count "<<count<<endl;
 }
};

int sample :: count;
```

```
void main()
{
 clrscr();
 sample s1,s2;
 s1.showcount();
 s2.showcount();
 s1.setdata();
 s2.setdata();
 s1.showdata();
 s2.showdata();
 sample :: showcount();
 s1.showcount();
 s2.showcount();
 getch();
}
```

OUTPUT

```
count 0
count 0
a= 1
a= 2
count 2
count 2
count 2
```

21. CPP Program to illustrate array of objects

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
class student
{
 int roll,age;
 char citizen[10];

 public:
 void setdata()
 {
  cout<<"Enter the Roll Number: ";
  cin>>roll;
  cout<<"Enter the age: ";
  cin>>age;
  cout<<"Enter the Country of the candidate: ";
  gets(citizen);
 }

 void showdata()
 {
  cout<<"\nRoll number: "<<roll<<"\nAge: "<<age<<"\nCountry: "<<citizen<<endl;
 }
};
```

```
void main()
{
 clrscr();
 student s[10];
 cout<<"Enter the number of the records: ";
 int n;
 cin>>n;
 cout<<endl;
 for(int i=0;i<n;i++)
 {
  cout<<".............Enter record "<<i+1<<".............\n";
   s[i].setdata();
  cout<<endl;
 }

 cout<<endl<<"You have entered following "<<i<<" Records"<<endl;
 for(i=0;i<n;i++)
 {
  cout<<endl<<".............Record "<<i+1<<".............";
  s[i].showdata();
 }
 getch();
}
```

OUTPUT

```
Enter the number of the records: 2
```

.....Enter record 1.....

Enter the Roll Number: 12

Enter the age: 15

Enter the Country of the candidate: India

.....Enter record 2.....

Enter the Roll Number: 15

Enter the age: 13

Enter the Country of the candidate: Russia

You have entered following 2 Records

.....Record 1.....

Roll number: 12

Age: 15

Country: India

.....Record 2.....

Roll number: 15

Age: 13

Country: Russia

22. CPP Program illustrating object as an argument

```
#include<iostream.h>
#include<conio.h>
class time
{
 int h,m;
 public:
 void settime(int a,int b)
 {
  h=a;
  m=b;
 }
 void showtime()
 {
  cout<<h<<"Hrs "<<m<<"Minutes";
 }

 void addtime(time t1,time t2)
 {
  h=(t1.h+t2.h+(t1.m+t2.m)/60);
  m=(t1.m+t2.m)%60;
 }
};

void main()
```

```
{
 time t1,t2,t3;
 clrscr();
 t1.settime(1,15);
 t2.settime(1,50);
 t3.addtime(t1,t2);
 t3.showtime();
 getch();
}
```

OUTPUT

```
3Hrs 5Minutes
```

23. CPP Program to illustrating Friend function

```
#include<iostream.h>
#include<conio.h>
class complex
{
 int real,imag;
 public:
 void setdata()
 {
  cout<<"\nEnter the value of real part and imaginary part respectively: \n";
  cin>>real>>imag;
 }

 void showdata()
 {
 cout<<real<<" + i("<<imag<<")";
 }
 friend complex addition(complex,complex);
};

complex addition(complex m,complex n)
{
 complex temp;
 temp.real=m.real+n.real;
 temp.imag=m.imag+n.imag;
 return temp;
```

```
}

void main()
{
 clrscr();
 complex c1,c2,c3;
 cout<<"Enter 1st complex number: ";
 c1.setdata();
 cout<<"Enter 2nd complex number: ";
 c2.setdata();
 cout<<"You have Entered\n";
 c1.showdata();
 cout<<endl;
 c2.showdata();

 cout<<"\nAddition is: ";
 c3=addition(c1,c2);
 c3.showdata();
 getch();
}
```

OUTPUT

Enter 1st complex number:

Enter the value of real part and imaginary part respectively:

10

-12

Enter 2nd complex number:

Enter the value of real part and imaginary part respectively:

100

10

You have Entered

10 + i(-12)

100 + i(10)

Addition is: 110 + i(-2)

24. CPP Program for illustrating friend function which friend of two classes

```
#include<iostream.h>
#include<conio.h>
class X;

class A
{
 int a;
 public:
 void setdata()
 {
  cin>>a;
 }
 friend int maximum(A,X);
};

class X
{
 int x;
 public:
 void setdata()
 {
  cin>>x;
 }
 friend int maximum(A,X);
```

```
};

int maximum(A m,X n)
{
 if(m.a>n.x)
  return m.a;
 else
  return n.x;
}
void main()
{
 clrscr();
 A a1;
 X x1;
 cout<<"Enter two elements for comparing\n";
 a1.setdata();
 x1.setdata();
 cout<<endl<<maximum(a1,x1)<<" is large";
 getch();
}
```

OUTPUT

```
Enter two elements for comparing
99
110
110 is large
```

25. CPP Program to illustrating constructor

```
#include<iostream.h>
#include<conio.h>
class sample
{
 int a;
 public:
 sample()   //default constructor
 {a=11;
 }
 void showdata()
 {
  cout<<"a="<<a;
 }
};

void main()
{
 clrscr();
 sample s1;
 s1.showdata();
 getch();
}
```

OUTPUT

```
a=11
```

26. CPP Program to illustrating constructor overloading

```
#include<iostream.h>
#include<conio.h>
class complex
{
 float x,y;
 public:

 complex()   //default constructor
 {}

 complex(float a)  //parameterised constructor
 {
  x=y=a;
 }

 complex(float real,float img)
 {
  x=real;
  y=img;
 }

 friend complex sum(complex,complex);
 friend void show(complex);
};
```

```
complex sum(complex c1,complex c2)
{
 complex c3;
 c3.x=c1.x+c2.x;
 c3.y=c1.y+c2.y;
 return c3;
}
void show(complex c)
{
 cout<<c.x<<" + i"<<c.y;
}
void main()
{
 clrscr();
 complex a(27,3.5);
 complex b(1.6);
 complex c;
 c=sum(a,b);
 show(a);
 cout<<endl;
 show(b);
 cout<<endl;
 show(c);
 getch();
}
```

OUTPUT

27 + i3.5

1.6 + i1.6

28.6 + i5.1

27. CPP Program for illustrating Destructor

```
#include<iostream.h>
#include<conio.h>
int count=0;
class text
{
 public:
 text()
 {
  count++;
  cout<<"\nCustructor msg: Object number "<<count<<" created";
 }

 ~text()
 {
  cout<<"\n\nDetructor msg: Object number "<<count<<" destroyed";
  count--;
 }
};

void main()
{
 clrscr();
 cout<<"Inside the main Block";
 cout<<"\ncreating First Object T1";
 text t1;
```

```
{
 cout<<"\nInside Block1";
 cout<<"\nCreating two more objects T2 and T3";
 text T2,T3;
 cout<<"\nLeaving Block1";
}
 cout<<"\nBlock Inside the main Block";
 getch();
}
```

OUTPUT

```
Inside the main Block
creating First Object T1
Custructor msg: Object number 1 created
Inside Block1
Creating two more objects T2 and T3
Custructor msg: Object number 2 created
Custructor msg: Object number 3 created
Leaving Block1

Detructor msg: Object number 3 destroyed

Detructor msg: Object number 2 destroyed
Block Inside the main Block

Detructor msg: Object number 1 destroyed
```

28. CPP Program for Operator overloading for unary operator

```
#include<iostream.h>
#include<conio.h>
class sample
{
 int a;
 public:
 void setdata()
 {
  cin>>a;
 }
 int operator ++()
 {
  int temp=++a;
  return temp;
 }
 void showdata()
 {
  cout<<"\na="<<a;
 }
};

void main()
{
 clrscr();
 sample s1;
```

```
int t;
cout<<"Enter any number\n";
s1.setdata();
cout<<"\n\nYou have entered";
s1.showdata();
cout<<"\n\nAfter pre-increment";
t=++s1;
cout<<"\nt="<<t;
s1.showdata();
getch();
}
```

OUTPUT

```
Enter any number
12

You have entered
a=12

After pre-increment
t=13
a=13
```

29. CPP Program for illustrating operator overloading for binary operator;

```
#include<iostream.h>
#include<conio.h>

class sample
{
 int a;
 public:

 void setdata()
 {
  cin>>a;
 }

 void showdata()
 {
  cout<<a;
 }

 sample operator +(sample s)
 {
  sample temp;
  temp.a=a+s.a;
  return temp;
 }
};
```

```
void main()
{
 clrscr();
 sample s1,s2,s3;
 cout<<"Enter any two numbers for addition\n";
 s1.setdata();
 s2.setdata();
 s3=s1+s2;
 cout<<"\nThe addition is: ";
 s3.showdata();
 getch();
}
```

OUTPUT

```
Enter any two numbers for addition
11
12

The addition is: 23
```

30. CPP Program for single Inheritance;

```
#include<iostream.h>
#include<conio.h>
class Parent
{
 public:
 int id_p;
};
class Child:public Parent
{
 public:
 int id_c;
};
void main()
{
 clrscr();
 Child obj1;
 obj1.id_c=7;
 obj1.id_p=9;

 cout<<"Child Id: "<<obj1.id_c;
 cout<<endl<<"Parent Id: "<<obj1.id_p;
 getch();}
```

OUTPUT

```
Child Id: 7
Parent Id: 9
```

31. CPP Program for Multilevel Inheritance

```
#include<iostream.h>
#include<conio.h>

class Base
{
 public:
 int x;

 void getdata()
 {
  cout<<"Enter value of x: ";
  cin>>x;
 }
};

class Derived1 :public Base
{
 public:
 int y;

 void readdata()
 {
  cout<<"\nEnter value of y: ";
  cin>>y;
 }
```

```
};

class Derived2 :public Derived1
{
 public:
  int z;
 void indata()
 {
  cout<<"Enter value of z: ";
  cin>>z;
 }
 void product()
 {
  cout<<"\nProduct= "<<x*y*z;
 }
};
void main()
{
 clrscr();
 Derived2 a;
 a.getdata();
 a.readdata();
 a.indata();
 a.product();
 getch();
}
```

OUTPUT

Enter value of x: 12

Enter value of y: 23

Enter value of z: 10

Product= 2760

32. CPP Program for Multiple Inheritance

```
#include<iostream.h>
#include<conio.h>

class A
{
 public:
 int x;

 void getx()
 {
  cout<<"Enter value of x: ";
  cin>>x;
 }
};

class B
{
 public:
 int y;

 void gety()
 {
  cout<<"Enter value of y: ";
  cin>>y;
 }
```

```
};

class C:public A,public B
{
 public:
 void sum()
 {
  cout<<"Sum= "<<x+y;
 }
};

void main()
{
 clrscr();
 C obj1;
 obj1.getx();
 obj1.gety();
 obj1.sum();
 getch();
}
```

OUTPUT

```
Enter value of x: 12
Enter value of y: 31
Sum= 4
```

33. CPP Program for Hierarchical Inheritance

```
#include<iostream.h>
#include<conio.h>

class A
{
 public:
 int x,y;

 void getdata()
 {
  cout<<"Enter value of x and y\n";
  cin>>x>>y;
 }
};

class B:public A
{
 public:
 void product()
 {
  cout<<"Product"<<x*y<<endl<<endl;
 }
};

class C:public A
```

```
{
 public:
 void sum()
 {
  cout<<"Sum= "<<x+y;
 }
};

void main()
{
 clrscr();
 B obj1;
 C obj2;
 obj1.getdata();
 obj1.product();
 obj2.getdata();
 obj2.sum();
 getch();
}
```

OUTPUT

Enter value of x and y

12

13

Product156

Enter value of x and y

12

76

Sum= 88

34. CPP Program for Hybrid Inheritance

```
#include<iostream.h>
#include<conio.h>

class A
{
 public:
 int x;
};

class B:public A
{
 public:
 B()
 {
  x=10;
 }
};

class C
{
 public:
 int y;
 C()
 {
  y=4;
```

```
 }
};

class D:public B,public C
{
 public:
 void sum()
 {
  cout<<"Sum= "<<x+y;
 }
};

void main()
{
 clrscr();
 D obj1;
 obj1.sum();
 getch();
}
```

OUTPUT

```
Sum= 14
```

35. CPP Program for Ambiguity Problem

```
#include<iostream.h>
#include<conio.h>
class GrandP
{
 int a;
};
class P1:public GrandP
{
 int x;
};
class P2:public GrandP
{
 int y;
};
class Child:public P1,P2  //Two copies of GrandP will be inherited
{
 int m;
};

void main()
{
 clrscr();
 Child c;
 getch();
}
```

36. CPP Program for Virtual Base class(Solving Ambiguity Problem)

```
#include<iostream.h>
#include<conio.h>
class GrandP
{
 int a;
};
class P1:public GrandP
{
 int x;
};
class P2:virtual public GrandP
{
 int y;
};
class Child:public virtual P1,P2  //Only one copy of GrandP will be inherited
{
 int m;
};

void main()
{
 clrscr();
 Child c;
 getch();
}
```

37. CPP Program illustrating pointer to an object

```
#include<iostream.h>
#include<conio.h>
class sample
{
 int a;
 public:
 void setdata()
 {
  cin>>a;
 }

 void showdata()
 {
  cout<<"a= "<<a;
 }
};

void main()
{
 clrscr();
 sample s1,*ptr;
 ptr=&s1;
 cout<<"Enter any data: ";
 ptr->setdata();
 cout<<"\nYou have Entered ";
```

```
 s1.showdata();
 getch();
}
```

OUTPUT

```
Enter any data: 11

You have Entered a= 11
```

38. CPP Program to illustrating This Pointer

```
#include<iostream.h>
#include<conio.h>

class Box
{
 int l,b,h;
 public:
 void setdimensions(int l,int b,int h)
 {
  this->l=l;
  this->b=b;
  this->h=h;
 }
 void showdimensions()
 {
  cout<<"L="<<l;
  cout<<endl<<"B="<<b;
  cout<<endl<<"H="<<h;
 }
};

void main()
{
 clrscr();
 Box b1;
```

```
 b1.setdimensions(112,101,20);
 b1.showdimensions();
 getch();
}
```

OUTPUT

```
L=112
B=101
H=20
```

39. CPP Program to illustrating pointer to derived class

```
#include<iostream.h>
#include<conio.h>
class Base
{
 public:
 void display()
 {
  cout<<"\nDisplay Base";
 }
};

class Derived: public Base
{
 public:
 void display()
 {
  cout<<"\nDisplay derived";
 }

 void fun()
 {
  cout<<endl<<"Display fun(), which is the member of only Derived class";
 }
};
```

```
void main()
{
 clrscr();
 Base B,*bptr;
 Derived D,*dptr;
 cout<<"\nbptr points to base";
 bptr=&B;
 bptr->display();
 cout<<"\ndptr points to derived";
 dptr=&D;
 dptr->fun();
 getch();
}
```

OUTPUT

```
bptr points to base
Display Base
dptr points to derived
Display fun(), which is the member of only Derived class
```

40. CPP Program to illustrating virtual function

```
#include<iostream.h>
#include<conio.h>
class Base
{
 public:
 void display()
 {
  cout<<"\nDisplay Base";
 }
 virtual void show()
 {
  cout<<"\nBase show";
 }
};

class Derived: public Base
{
 public:
 void display()
 {
  cout<<"\nDisplay derived";
 }

 void show()
 {
```

```
  cout<<"\nShow derived";
 }
};

void main()
{
 clrscr();
 Base B;
 Derived D;
 Base *bptr;
 cout<<"\nbptr points to base";
 bptr=&B;
 bptr->display();
 bptr->show();
 cout<<"\nbptr points to derived";
 bptr=&D;
 bptr->display();
 bptr->show();
 getch();
}
```

OUTPUT

bptr points to base

Display Base

Base show

bptr points to derived

Display Base

Show derived

41. CPP Program illustrating Abstract Class

```
#include<iostream.h>
#include<conio.h>
class student
{
 int roll_no;
 public:
 void getdata(int n)
 {
  int roll_no=n;
 }

 void putdata()
 {
  cout<<"Roll No="<<roll_no<<endl;
 }
 virtual void getmarks(float,float)=0;
 virtual void putmarks()=0;
};

class engineering: public student
{
 float sub1,sub2;
 public:
 void getmarks(float m,float n)
 {
```

```
 sub1=m;
 sub2=n;
 }
void putmarks()
 {
  cout<<"sub1="<<sub1<<endl;
  cout<<"sub2="<<sub2<<endl;
 }
};

class medical:public student
{
 float sub1,sub2;
 public:

 void getmarks(float m,float n)
 {
  sub1=m;
  sub2=n;
 }

 void putmarks()
 {
  cout<<"sub1="<<sub1<<endl;
  cout<<"sub2="<<sub2<<endl;
 }
```

```
};
void main()
{
 clrscr();
 student *p;
 engineering e;
 medical m;
 p=&e;
 e.getdata(102);
 p->putdata();
 p->getmarks(50.6,60.8);
 p->putmarks();
 p=&m;
 p->getmarks(90.4,89.7);
 p->putmarks();
 getch();
}
```

OUTPUT

```
Roll No=1
sub1=50.599998
sub2=60.799999
sub1=90.400002
sub2=89.699997
```

42. CPP Program to write data in a File

```
#include<iostream.h>

#include<fstream.h>

#include<conio.h>


void main()

{

 ofstream fout;

 fout.open("f1.txt");

 getch();d

 fout<<"Hello";

 getch();

 fout.close();

 getch();

}
```

43. CPP Program to read data from a File

```
#include<iostream.h>
#include<fstream.h>
#include<conio.h>

void main()
{
 clrscr();
 ifstream fin;
 char ch;
 fin.open("f1.txt");
 fin>>ch;
 while(!fin.eof())
 {
  cout<<ch;
  fin>>ch;
 }
 fin.close();
 getch();
}
```

OUTPUT

Hello